

# Modeling and Optimal Control for Two-Wheeled Self-Balancing Robot

Quoc-Thinh Do <sup>1</sup>, Van-Thanh Tran <sup>2,\*</sup>, Minh-Thai Ngo <sup>3</sup>, Minh-Quan Tran <sup>4</sup>, Quan-Linh Thiem <sup>5</sup>, Hoang-Son Nguyen <sup>6</sup>,  
Ba-Khoi Pham <sup>7</sup>, Nguyen-Phuoc-An Phan <sup>8</sup>, Duy-Hieu Nguyen <sup>9</sup>, Duc-Hoc Nguyen <sup>10</sup>, Van-Hoc Nguyen <sup>11</sup>,  
Ho-Minh-Quang Tran <sup>12</sup>, Thi-Hong-Lam Le <sup>13</sup>

<sup>1,2,3,4,5,6,7,8,9,10,11,12,13</sup> Ho Chi Minh City University of Technology and Education (HCMUTE), Vietnam

Email: <sup>1</sup> 19151293@student.hcmute.edu.vn, <sup>2</sup> 19151287@student.hcmute.edu.vn, <sup>3</sup> 19161161@student.hcmute.edu.vn,

<sup>4</sup> 20142070@student.hcmute.edu.vn, <sup>5</sup> 19145070@student.hcmute.edu.vn, <sup>6</sup> 19142090@student.hcmute.edu.vn,

<sup>7</sup> 20161215@student.hcmute.edu.vn, <sup>8</sup> 20142273@student.hcmute.edu.vn, <sup>9</sup> 20145502@student.hcmute.edu.vn,

<sup>10</sup> 20142336@student.hcmute.edu.vn, <sup>11</sup> 20142335@student.hcmute.edu.vn, <sup>12</sup> 20145590@student.hcmute.edu.vn,

<sup>13</sup> lamth@hcmute.edu.vn

\*Corresponding Author

**Abstract**—The two-wheeled self-balancing robot based on an inverted pendulum model is a nonlinear object with uncertain parameters that are difficult to control with 6 state variables. This is a multiple input-multiple output (MIMO) under-actuated system that is very complex and causes many challenges for the operator. This paper analyzed the mathematical equation of a two-wheeled self-balancing robot vehicle system. Then, the Linear Quadratic Regulator (LQR) control is applied to the system through simulation on Matlab/Simulink and experiment. The results show that the LQR algorithm has been successfully applied in many moving cases.

**Keywords**—Two-Wheeled Self-Balancing Robot; MIMO; Optimal Control; LQR Control; Under-Actuated System

## I. OVERVIEW

One of the research directions of the automatic control subject is the control algorithms. The two-wheeled self-balancing robot is one of the control topics related to DC motors [1]-[3]. This is a relatively classic control object in research laboratories. However, most studies are only based on simple mathematical equations with the system input being left or right wheel torque and only completed on simulation [4]-[6], while in the real model, the input is the voltage supplied to the left or right wheel motor (more complex system mathematical equation). Besides, studies applying PID, Fuzzy algorithms that are not based on mathematical models will produce system stability results that are not guaranteed by mathematics [7]-[9]. Therefore, establishing a detailed mathematical model with voltage control input is necessary. Once the detailed mathematical equation has been established, an optimal control algorithm (LQR) is the best solution to control the system because the stability of the system is guaranteed by solving the Riccati equation [10]-[12]. However, in these studies, the only simulation result is shown, and control inputs are moments of the motor. In [13], and [14], experimental models are presented. However, only stabilizing LQR control at equilibrium point is proved in that research. The survey of turn right-left is not completed. In [15], a LQG (developed from LQR method) is also proved in balancing at equilibrium point of this model. But the survey of other cases of turning left-right is not mentioned.

In this paper, the authors conducted verification through simulations on Matlab/Simulink and experiments. The obtained results show that the LQR algorithm has been successfully applied to a two-wheeled self-balancing robot in cases of balancing, turning right, and turning left.

The next sections of the article are presented as follows: Section II will discuss how to model a self-balancing two-wheeled vehicle system. Next, section III will present the optimal control algorithm applied to this vehicle system. Then, sections IV and V present the simulation and experimental control results, respectively. Finally, section VI provides conclusions.

## II. MATHEMATICAL MODEL

The diagram and reference system of the self-balancing two-wheeled vehicle system are shown in Fig. 1 and Fig. 2, respectively. In particular, components of system are listed in Table 1.

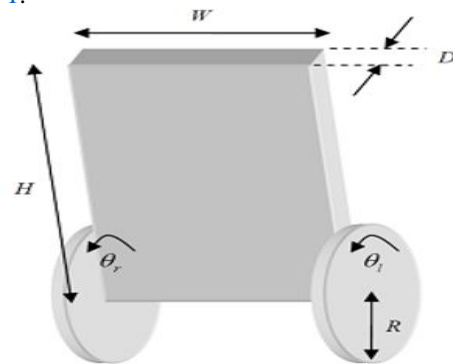


Fig. 1. Describes the two-wheeled self-balancing robot

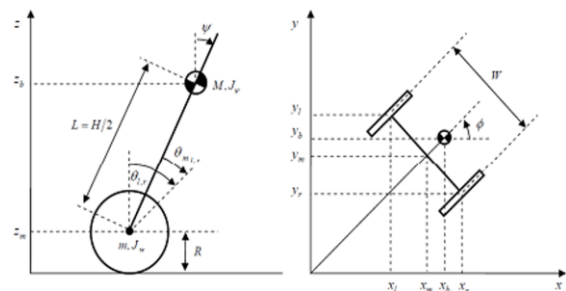


Fig. 2. Two-wheeled self-balancing robot viewed from top and side

Table 1. Model parameters

Notation	Unit	Meaning
M	Kg	Mass of the wheel
m	Kg	Mass of the robot
R	m	Radius of the wheel
W	m	Width of the robot
D	m	Horizontal of the robot
H	m	Height of the robot
L	m	Distance from robot center of gravity to wheel axis
$f_w$		Coefficient of friction between the wheel and the moving surface
$f_m$		Friction coefficient between robot and DC motor
$J_m$	kg. m2	Moment of inertia of DC motor
$R_m$	Ohm	DC motor resistance
$K_b$	V sec/rad	EMF coefficient of DC motor
$K_t$	Nm/A	Torque of DC motor
N		Deceleration ratio
G	m/s2	Gravitational acceleration
$\theta$	Rad	Average angle of left and right wheels
$\theta_{l,r}$	Rad	Angle of left and right wheels
$\psi$	Rad	Tilt angle of the robot body
$\phi$	Rad	Rotation angle of the robot
$x_l, y_l, z_l$	M	Left wheel coordinates
$x_r, y_r, z_r$	M	Right wheel coordinates
$x_m, y_m, z_m$	M	Average coordinates
$F\theta, F\psi, F\phi$	Nm	Moments act in different directions
$F_{l,r}$	Nm	Starting moment of left and right wheel motor

We use the Euler-Lagrange method to build the dynamic model. Supposing that at time  $t = 0$ , the robot moves in the positive x-axis direction, we have the equations below. The average translation angle of the two wheels and the rotation angle of the robot are determined as follows:

$$\begin{pmatrix} \theta \\ \phi \end{pmatrix} = \begin{pmatrix} \frac{1}{2}(\theta_l + \theta_r) \\ \frac{R}{W}(\theta_l - \theta_r) \end{pmatrix} \quad (1)$$

The average coordinates of the robot in the frame reference are:

$$\begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix} = \begin{bmatrix} \int \dot{x}_m \\ \int \dot{y}_m \\ R \end{bmatrix} \quad (2)$$

and

$$\begin{bmatrix} \dot{x}_m \\ \dot{y}_m \end{bmatrix} = \begin{bmatrix} R\dot{\theta} \cos \phi \\ R\dot{\theta} \sin \phi \end{bmatrix} \quad (3)$$

Coordinates of the left wheel in the frame reference are:

$$\begin{bmatrix} x_l \\ y_l \\ z_l \end{bmatrix} = \begin{pmatrix} x_m - \frac{W}{2} \sin \phi \\ y_m + \frac{W}{2} \cos \phi \\ Z_M \end{pmatrix} \quad (4)$$

Coordinates of the right wheel in the frame reference are:

$$\begin{bmatrix} x_r \\ y_r \\ z_r \end{bmatrix} = \begin{pmatrix} x_m + \frac{W}{2} \sin \phi \\ y_m - \frac{W}{2} \cos \phi \\ Z_M \end{pmatrix} \quad (5)$$

Coordinates of the center of symmetry between the two motors in the frame reference are:

$$\begin{bmatrix} x_b \\ y_b \\ z_b \end{bmatrix} = \begin{bmatrix} x_m + L \sin \psi \cos \phi \\ y_m L \sin \psi \sin \phi \\ z_m + L \cos \phi \end{bmatrix} \quad (6)$$

Kinetic energy of translational motion is

$$T_1 = \frac{1}{2}m(\dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2) + \frac{1}{2}m(\dot{x}_l^2 + \dot{y}_l^2 + \dot{z}_l^2) + \frac{1}{2}m(\dot{x}_b^2 + \dot{y}_b^2 + \dot{z}_b^2) \quad (7)$$

Kinetic energy of rotation is

$$T_2 = \frac{1}{2}J_w\dot{\theta}_l^2 + \frac{1}{2}J_w\dot{\theta}_r^2 + \frac{1}{2}J_\psi\dot{\psi}^2 + \quad (8)$$

$$\frac{1}{2}J_\phi\dot{\phi}^2 + \frac{1}{2}n^2J_m(\dot{\theta}_l - \dot{\psi})^2 + \frac{1}{2}n^2J_m(\dot{\theta}_r - \dot{\psi})^2$$

Where  $\frac{1}{2}n^2J_m(\dot{\theta}_r - \dot{\psi})^2$ ;  $\frac{1}{2}n^2J_m(\dot{\theta}_l - \dot{\psi})^2$  is the rotational kinetic energy of the left and right motor armatures. Potential energy is:

$$U = mgz_l + mgz_r + mgz_b \quad (9)$$

Lagrangian operator is:

$$L = T_1 + T_2 - U \quad (10)$$

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\psi}} \right) - \frac{\partial L}{\partial \psi} = F_\psi \quad (11)$$

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\phi}} \right) - \frac{\partial L}{\partial \phi} = F_\phi \quad (12)$$

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\phi}} \right) - \frac{\partial L}{\partial \phi} = F_\phi \quad (13)$$

Derivating L with respect to the variables, we obtain

$$[(2m + M)R^2 + 2J_w + 2n^2J_m]\ddot{\theta} + \quad (14)$$

$$(MLR \cos \psi - 2n^2J_m)\dot{\psi} - MLR\dot{\psi}^2 \sin \psi = F_\theta$$

$$(MLR \cos \psi - 2n^2J_m)\ddot{\theta} + (ML^2 + J_\psi + 2n^2J_m)\ddot{\psi} - \quad (15)$$

$$MgL \sin \psi - ML^2\dot{\phi}^2 \sin \psi \cos \psi = F_\psi$$

$$\left[ \frac{1}{2}mW^2 + J_\phi + \frac{W^2}{2R^2}(J_w + n^2J_m) + ML^2 \sin^2 \psi \right] \ddot{\phi} + \quad (16)$$

$$2ML^2\dot{\psi}\dot{\phi} \sin \psi \cos \psi = F_\phi$$

Dynamic torques produced by DC motors are:

$$\begin{bmatrix} F_\theta \\ F_\psi \\ F_\phi \end{bmatrix} = \begin{bmatrix} F_l + F_r \\ F_\psi \\ \frac{W}{2R}(F_l - F_r) \end{bmatrix} \quad (17)$$

Where  $F_l = nK_t i_l + f_m(\dot{\psi} - \dot{\theta}_l) - f_w\dot{\theta}_l$ ;  $F_r = nK_t i_r + f_m(\dot{\psi} - \dot{\theta}_r) - f_w\dot{\theta}_r$ ;  $F_\psi = -nK_t i_l - nK_t i_r - f_m(\dot{\psi} - \dot{\theta}_l) - f_m(\dot{\psi} - \dot{\theta}_r)$

Using the Pulse Width Modulation (PWM) method to control the motor, we need to convert from torque to motor voltage through this formula:

$$L_m \dot{i}_{l,r} = v_{l,r} + K_b(\dot{\psi} - \dot{\theta}_{l,r}) - R_m i_{l,r} \quad (18)$$

Assuming the armature inductance is relatively small (close to 0), it can be ignored and deduced:

$$\dot{i}_{l,r} = \frac{v_{l,r} + k_b(\dot{\psi} - \dot{\theta}_{l,r})}{R_m} \quad (19)$$

From there, torques are generated as:

$$F_\theta = \alpha(v_l + v_r) - 2(\beta + f_w)\dot{\theta} + 2\beta\dot{\psi} \quad (20)$$

$$F_\psi = -\alpha(v_l + v_r) + 2\beta\dot{\theta} - 2\beta\dot{\psi} \quad (21)$$

$$F_\phi = \frac{W}{2R}\alpha(v_r - v_l) - \frac{W^2}{2R^2}(\beta + f_w)\dot{\phi} \quad (22)$$

Where  $\alpha = \frac{nk_t}{R_m}$  và  $\beta = \frac{nk_t k_b}{R_m} + f_m$  Thence, we obtain the dynamic equations describing the robot's motion as follows:

$$\begin{aligned} & [(2m + M)R^2 + 2J_w + 2n^2J_m]\ddot{\theta} \\ & + (MLR \cos \psi - 2n^2J_m)\ddot{\psi} \\ & - MLR\dot{\psi}^2 \sin \psi \\ & = \alpha(v_l + v_r) - 2(\beta + f_w)\dot{\theta} + 2\beta\dot{\psi} \end{aligned} \quad (23)$$

$$\begin{aligned} & (MLR \cos \psi - 2n^2J_m)\ddot{\theta} + (ML^2 + J_\psi + 2n^2J_m)\ddot{\psi} \\ & - MgL \sin \psi - ML^2\dot{\phi}^2 \sin \psi \cos \psi \\ & = -\alpha(v_l + v_r) + 2\beta\dot{\theta} - 2\beta\dot{\psi} \end{aligned} \quad (24)$$

$$\begin{aligned} & \left[ \frac{1}{2}mW^2 + J_\phi + \frac{W^2}{2R^2}(J_w + n^2J_m) + ML^2 \sin^2 \psi \right] \ddot{\phi}^2 \\ & + 2ML^2\dot{\psi}\dot{\phi} \sin \psi \cos \psi \\ & = \frac{W}{2R}\alpha(v_r - v_l) - \frac{W^2}{2R^2}(\beta + f_w)\dot{\phi} \end{aligned} \quad (25)$$

### III. OPTIMAL CONTROL

From (23), (24), (25), we have a system of equations describing a balanced two-wheeled vehicle system of the following form as:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f_1(x_1, x_2, x_3, x_4, x_5, x_6, v_l, v_r) \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = f_2(x_1, x_2, x_3, x_4, x_5, x_6, v_l, v_r) \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = f_3(x_1, x_2, x_3, x_4, x_5, x_6, v_l, v_r) \end{cases} \quad (26)$$

Where

$$x = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6]^T = [\theta \ \dot{\theta} \ \psi \ \dot{\psi} \ \phi \ \dot{\phi}]^T \quad (27)$$

If we choose the working point as:

$$x_0 = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T; u_0 = [0 \ 0]^T \quad (28)$$

Then, we can linearize the system in form:

$$\dot{x} = Ax + Bu \quad (29)$$

Where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{\partial h_1}{\partial x_1} & \frac{\partial h_1}{\partial x_2} & \frac{\partial h_1}{\partial x_3} & \frac{\partial h_1}{\partial x_4} & \frac{\partial h_1}{\partial x_5} & \frac{\partial h_1}{\partial x_6} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{\partial h_2}{\partial x_1} & \frac{\partial h_2}{\partial x_2} & \frac{\partial h_2}{\partial x_3} & \frac{\partial h_2}{\partial x_4} & \frac{\partial h_2}{\partial x_5} & \frac{\partial h_2}{\partial x_6} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{\partial h_3}{\partial x_1} & \frac{\partial h_3}{\partial x_2} & \frac{\partial h_3}{\partial x_3} & \frac{\partial h_3}{\partial x_4} & \frac{\partial h_3}{\partial x_5} & \frac{\partial h_3}{\partial x_6} \end{bmatrix}_{x=x_0, u=u_0}; B = \begin{bmatrix} 0 & 0 \\ \frac{\partial h_1}{\partial v_l} & \frac{\partial h_1}{\partial v_r} \\ 0 & 0 \\ \frac{\partial h_2}{\partial v_l} & \frac{\partial h_2}{\partial v_r} \\ 0 & 0 \\ \frac{\partial h_3}{\partial v_l} & \frac{\partial h_3}{\partial v_r} \end{bmatrix}_{x=x_0, u=u_0}$$

At this point, we have the following weight matrices:

$$Q = \begin{bmatrix} Q1 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q2 & 0 & 0 & 0 & 0 \\ 0 & 0 & Q3 & 0 & 0 & 0 \\ 0 & 0 & 0 & Q4 & 0 & 0 \\ 0 & 0 & 0 & 0 & Q5 & 0 \\ 0 & 0 & 0 & 0 & 0 & Q6 \end{bmatrix}; R = \begin{bmatrix} R1 & 0 \\ 0 & R2 \end{bmatrix} \quad (30)$$

With parameters Q1 to Q6, R1 and R2, we adjust the LQR controller. Parameters Q1, Q2, Q3, Q4, Q5, and Q6 are the optimal weights for variables  $\theta$ ,  $\dot{\theta}$ ,  $\psi$ ,  $\dot{\psi}$ ,  $\phi$ ,  $\dot{\phi}$ . A control signal for advance angle and a control signal for the rotation angle. If we want a state variable to have priority in control, we make that corresponding weight parameter relatively larger than the other weight parameters. As for the R matrix, the selection of R1 and R2 must be equal because the two wheels are proportional to each other. On the other hand, if the R matrix parameter is chosen too large, the two-wheel control signal will be strong, and responsive but sometimes unstable. If R matrix parameter is small, the model runs more smoothly. However, in some cases, it does not respond quickly and strongly enough to control the system. After selecting the corresponding control parameters, we proceed to build the feedback parameter K with the control signal.

$$u = -Kx \quad (31)$$

Parameter K is calculated based on the Riccati equation. However, Matlab supports the lqr() function to calculate the parameter K when the matrices A, B, Q, and R as

$$K = \text{lqr}(A, B, Q, R) \quad (32)$$

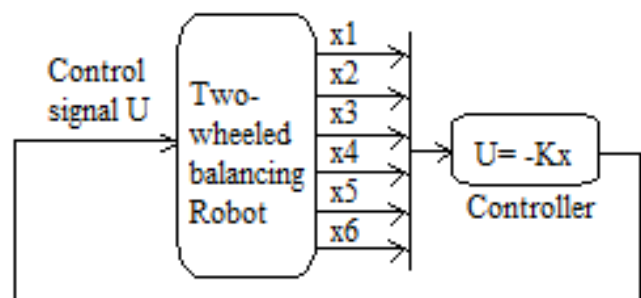


Fig. 3. LQR control for two-wheeled self-balancing robot

### IV. SIMULATION

The model parameters are selected as follows:  $m=0.037$ ;  $M=0.037$ ;  $R=0.035$ ;  $W=0.15$ ;  $D=0.075$ ;  $H=0.15$ ;  $L=0.075$ ;  $f_w=0.18$ ;  $f_m=0.002$ ;  $J_m=0.01$ ;  $R_m=50$ ;  $K_b=0.043$ ;  $K_t=0.166$ ;  $n=21.3$ ;  $g=9.81$ . The weight matrices are chosen as follows:

The LQR structure in Fig. 3 can be described in Matlab/Simulink program of simulation as in Fig. 4. In

block that simulates the two-wheeled self-balancing cart, smaller blocks in Fig. 5 describe this system from equations (23)-(25). The simulation results are shown in Fig. 6, Fig. 7, and Fig. 8.

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}; R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (33)$$

We can calculate the K matrix as follows:

$$K = \begin{bmatrix} -0.7 & 175.5 & -250.7 & -181.6 & -0.7 & -9.6 \\ -0.7 & 175.5 & -250.7 & -181.6 & 0.7 & 9.6 \end{bmatrix} \quad (34)$$

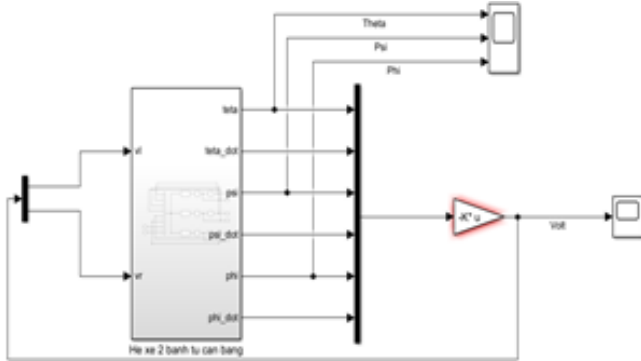


Fig. 4. LQR control simulation

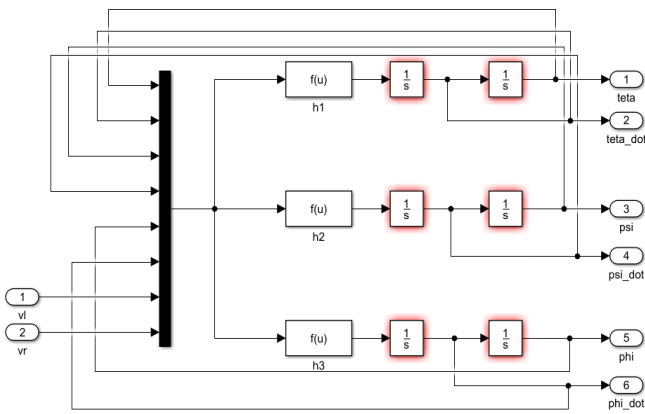


Fig. 5. Simulation of two-wheeled self-balancing robot

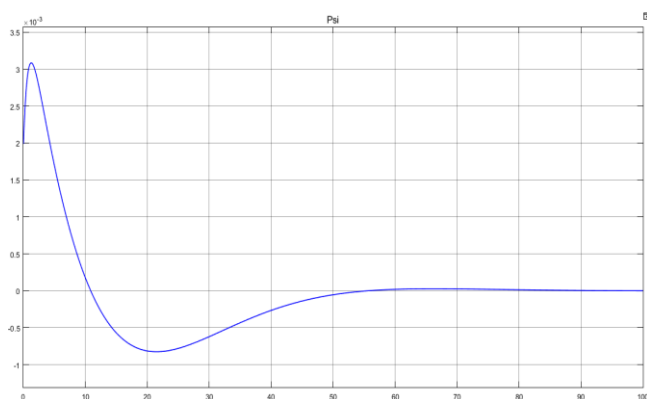


Fig. 6. Tilt angle (rad)

From results from Fig. 6, Fig. 7, and Fig. 8, the vehicle's state always follows the set signal, and the LQR controller responds quite well. Vibration is less than 0.1 degree and settling time is 60s. This proves that we can control the car

in many different situations such as: driving the car forward and backward, turning left, and turning right.

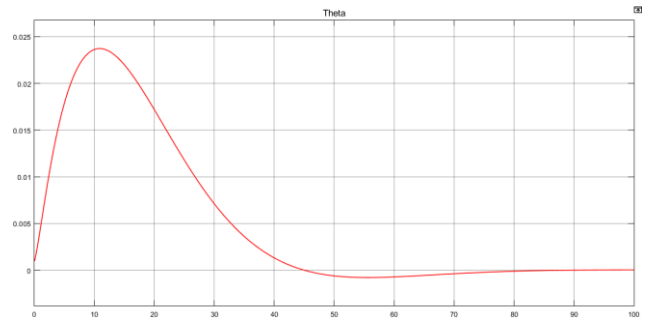


Fig. 7. Wheel angle (rad)

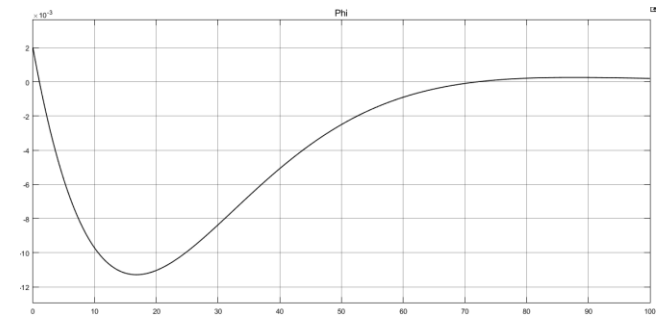


Fig. 8. Rotation angle (rad)

## V. EXPERIMENTS

The experimental model is shown in Fig. 9. The model includes:

1. Module HC-05.
2. Arduino Mega 2560.
3. LM298.
4. LM2596 DC Step Down Converter.
5. Rechargeable batteries.
6. DC motor.

Optimizing the weight matrices Q and R can be done by applying genetic algorithm (GA). In this project, the team adjusted the K matrix parameters many times and produced the following results. After adjustment, the K matrix is:

$$K = \begin{bmatrix} 21 & 42 & 3000 & 105 & 380 & 12 \\ 21 & 42 & 3000 & 105 & -380 & -12 \end{bmatrix} \quad (35)$$

### A. Stable Control at Balanced Position

The tilt angle fluctuates between -3 and 3 degrees, and the rotation angle and wheel angle fluctuate around the balance position. The two motor speeds are equivalent. The vehicle is stable and balanced. In Fig. 10, the body of robot is kept balanced at upward position with vibration is around -5 degrees and 5 degrees. The robot has self-balanced for more than 60s.

In Fig. 11, the wheel angle oscillates around the equilibrium position with vibrations of about 100 degrees and 150 degrees. Robot is stabilized at position which is in front of the equilibrium position on plane. In Fig. 12, the rotation angle oscillates around the equilibrium position with vibrations of about -2 degrees and 2 degrees. The cart is kept no rotating. Thence, in this experiment, robot is controlled well by PID method to be balanced at equilibrium point.

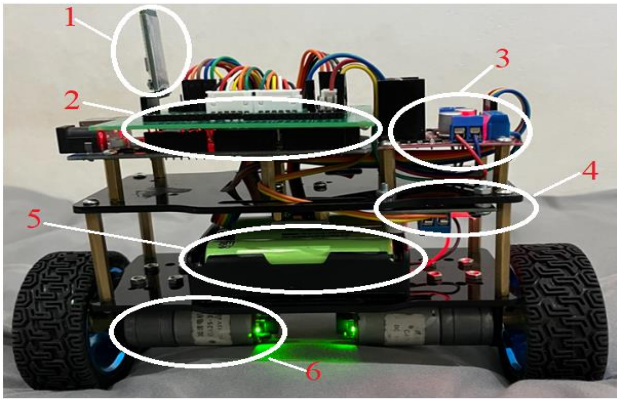


Fig. 9. Two-wheeled self-balancing robot model

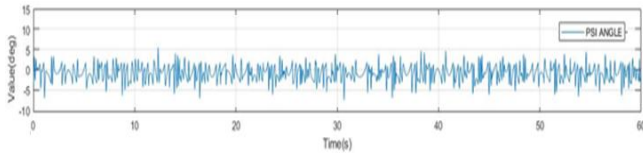


Fig. 10. Tilt angle (degrees)

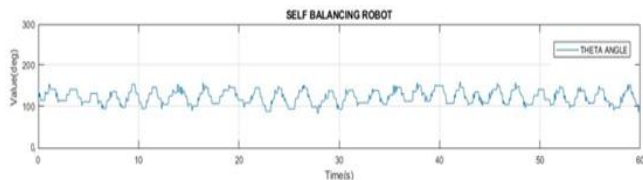


Fig. 11. Wheel angle (degrees)



Fig. 12. Rotation angle (degrees)

### B. Control with a Non-Zero Setting

Experiment 1: Forward control. During the period from 53s to 66s, we change the set value to let the car move forward. Fig. 14 shows that the wheel angle points upward but still maintains the vehicle's tilt and rotation angle around the equilibrium position as shown in Fig. 13 and Fig. 15.



Fig. 13. Tilt angle (degrees)

In Fig. 13, the body of robot is kept balanced at upward position with vibration is around -5 degrees and 5 degrees. In Fig. 14, in the period from 53s to 66s, the set value is changed, the wheel angle is directed upward.

In Fig. 15, the rotation angle oscillates around the equilibrium position with vibrations of about -2 degrees and 2 degrees. Experiment 2: Backward control. During the period from 44s to 65s, we change the set value to drive the vehicle in reverse. Fig. 17 shows that the wheel angle points downward but still maintains the vehicle's tilt and rotation

angle around the equilibrium position as shown in Fig. 16 and Fig. 18.

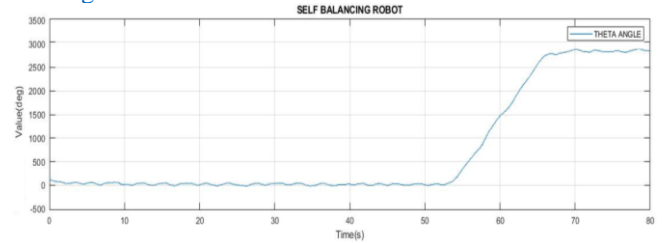


Fig. 14. Wheel angle (degrees)

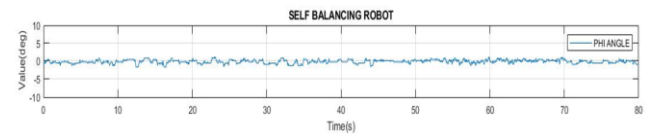


Fig. 15. Rotation angle (degrees)



Fig. 16. Tilt angle (degrees)

In Fig. 16, the body of robot is kept balanced at upward position with vibration is around -6 degrees and 6 degrees.

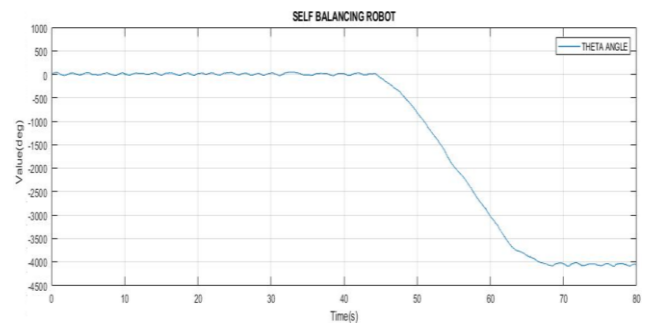


Fig. 17. Wheel angle (degrees)

In Fig. 17, in the period from 44s to 65s, the set value is changed, the wheel angle is directed downward.



Fig. 18. Rotation angle (degrees)

In Fig. 18, the rotation angle oscillates around the equilibrium position with vibrations of about -2 degrees and 2 degrees. Experiment 3: Control turning left. At the time of impact on the set value (from 34s to 38s), Fig. 21 shows that the rotation angle is downward but still maintained the tilt angle and wheel angle around the equilibrium position as Fig. 19 and Fig. 20. In Fig. 19, the body of robot is kept balanced at upward position with vibration is around -5 degrees and 6 degrees.

In Fig. 20, during the period from 34s to 38s, the set value is changed, the wheel angle is directed upward and the

body of robot is kept balanced with vibration is around 5 degrees and 75 degrees. In Fig. 21, in the period from 34s to 38s, the set value is changed, the wheel angle is directed downward. Robot is controlled well to turn left by changing rotational angle.

Experiment 4: Control turning right At the time of impact on the set value (from 33s to 37s), Fig. 24 shows that the rotation angle is upward but still maintained the tilt angle and wheel angle around the equilibrium position as shown in Fig. 22 and Fig. 23. In Fig. 22, the body of robot is kept balanced at upward position with vibration is around -5 degrees and 5 degrees.



Fig. 19. Tilt angle (degrees)

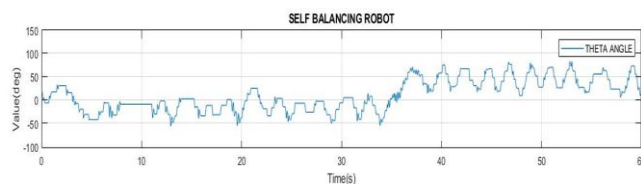


Fig. 20. Wheel angle (degrees)

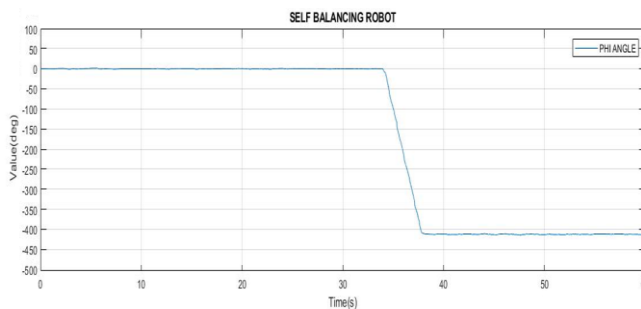


Fig. 21. Rotation angle (degrees)

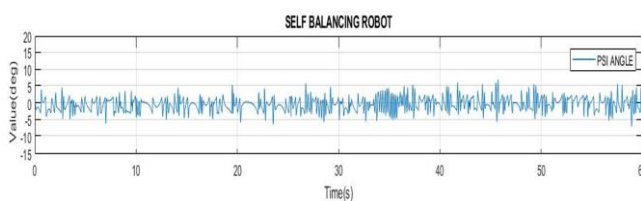


Fig. 22. Tilt angle (degrees)

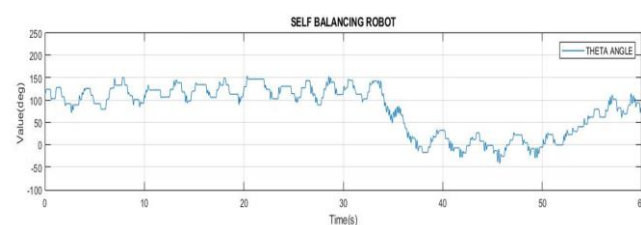


Fig. 23. Wheel angle (degrees)

In Fig. 23, during the period from 33s to 37s, the set value is changed, the wheel angle is directed downward and the body of robot is kept balanced with vibration is around -

25 degrees and 25 degrees. In Fig. 24, in the period from 34s to 38s, the set value is changed, the wheel angle is directed upward. Robot is controlled well to turn right by changing rotational angle.

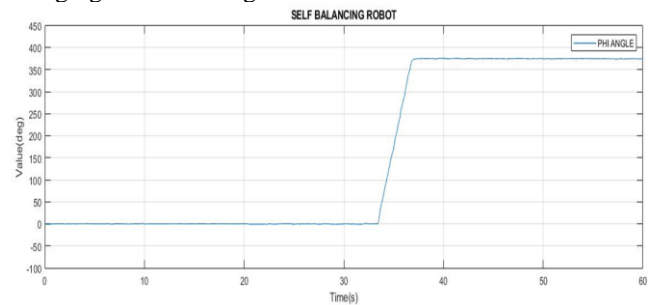


Fig. 24. Rotation angle (degrees)

## VI. CONCLUSION

In this project, an LQR controller is successfully designed for a self-balancing two-wheeled vehicle system - a complex and familiar MIMO nonlinear system in laboratories. The authors also successfully applied the simulation results to a realistic two-wheeled balance vehicle model using Arduino Mega 2560 microcontroller. LQR controller was tested through many cases: balancing at equilibrium point, positioning forward and backward, turning right, and turning left. It can be seen that in most cases, the LQR controller is successful in stabilizing the system and gives quite good control results. Accordingly, self-balancing two-wheeled vehicle model is a suitable model to verify and compare control algorithms, especially nonlinear algorithms.

## REFERENCES

- [1] P. Chevrel, L. Sicot and S. Siala, "Switched LQ controllers for DC motor speed and current control: a comparison with cascade control," *PESC Record. 27th Annual IEEE Power Electronics Specialists Conference*, vol. 1, pp. 906-912, 1996, <https://doi.org/10.1109/PESC.1996.548689>.
- [2] M. Ruderman, J. Krettek, F. Hoffmann, T. Bertram, "Optimal state space control of DC motor," *IFAC Proceedings Volumes*, vol 41, no. 2, pp 5796-5801, 2008, <https://doi.org/10.3182/20080706-5-KR-1001.00977>.
- [3] Z. Xiang and W. Wei1, "Design of DC motor position tracking system based on LQR," *Journal of Physics: Conference Series*, vol. 1887, no. 1, p. 012052, 2021, <https://doi.org/10.1088/1742-6596/1887/1/012052>.
- [4] A. N. K. Nasir, M. A. Ahmad, R. R. Ismail, "The Control of a Highly Nonlinear Two-wheels Balancing Robot: A Comparative Assessment between LQR and PID-PID Control Schemes," *International Journal of Mechanical and Mechatronics Engineering*, vol. 4, no. 10, pp. 942-947, 2010, <https://doi.org/10.5281/zenodo.1084448>.
- [5] N. N. Son, H. P. H. Anh, "Adaptive Backstepping Self-balancing Control of a Two-wheel Electric Scooter," *International Journal of Advanced Robotic Systems*, vol. 11, no. 10, 2014, <https://doi.org/10.5772/59100>.
- [6] M. Yue, W. Sun, P. Hu, "Sliding Mode Robust Control for Two-wheeled Mobile Robot with Lower Center of Gravity," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 2, 2011, <http://ijicic.org/09-0955-1.pdf>.
- [7] S. Miasa, M. Al-Mjali, A. Al-Haj Ibrahim and T. A. Tutunji, "Fuzzy control of a two-wheel balancing robot using DSPIC," *2010 7th International Multi- Conference on Systems, Signals and Devices*, pp. 1-6, 2010, <https://doi.org/10.1109/SSD.2010.5585525>.
- [8] A. Unluturk, O. Aydogdu and U. Guner, "Design and PID control of two wheeled autonomous balance robot," *2013 International*

- Conference on Electronics, Computer and Computation (ICECCO)*, pp. 260-264, 2013, <https://doi.org/10.1109/ICECCO.2013.6718278>.
- [9] C. Qiu, Y. Huang, "The Design of Fuzzy Adaptive PID Controller of Two-Wheeled Self-Balancing Robot," *International Journal of Information and Electronics Engineering*, vol. 5, no. 3, pp. 193-197, 2015, <https://doi.org/10.7763/IJIEE.2015.V5.529>.
- [10] N. Yang, J. Tang, Y. B. Wong, Y. Li and L. Shi, "Linear Quadratic Control of Positive Systems: A Projection-Based Approach," *IEEE Transactions on Automatic Control*, vol. 68, no. 4, pp. 2376-2382, 2023, <https://doi.org/10.1109/TAC.2022.3172237>.
- [11] M. O. Asali, F. Hadary, B. W. Sanjaya, "Modeling, Simulation, and Optimal Control for Two-Wheeled Self-Balancing Robot," *International Journal of Electrical and Computer Engineering*, vol. 7, no. 4, p. 2008, 2017, <http://doi.org/10.11591/ijece.v7i4.pp2008-2017>.
- [12] R. Petcu and G. -D. Andreescu, "Two-Wheeled Inverted-Pendulum Self-Balancing Robot with LQR or PID Control," *2022 IEEE 10th Jubilee International Conference on Computational Cybernetics and Cyber-Medical Systems (ICCC)*, pp. 000345-000350, 2022, <https://doi.org/10.1109/ICCC202255925.2022.9922862>.
- [13] D. O. Neacșu and A. Sirbu, "Design of a LQR-Based Boost Converter Controller for Energy Savings," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 7, pp. 5379-5388, 2020, <https://doi.org/10.1109/TIE.2019.2934062>.
- [14] Feriyonika, A. Hidayat, "Balancing Control of Two-Wheeled Robot by using Linear Quadratic Gaussian (LQG)," *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 12 no. 3, pp. 55-59, 2020, <https://jtec.utem.edu.my/jtec/article/view/5619>.
- [15] A. Amin, M. A. Fouz and A. Elsawaf, "LQR Stability Control of Scissor Pair Gyro-Stabilized Two Wheel Robot," *2023 5th International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE)*, pp. 1-7, 2023, <https://doi.org/10.1109/REEPE57272.2023.10086769>.